

УДК 519.622

## Application of Explicit Methods with Extended Stability Regions for Solving Stiff Problems

**Eugeny A. Novikov\***

Institute of Computational Modeling SB RAS  
Akademgorodok, 50/44, Krasnoyarsk, 660036  
Russia

**Mikhail V. Rybkov†**

Institute of Mathematics and Computer Science  
Siberian Federal University  
Svobodny, 79, Krasnoyarsk, 660041  
Russia

---

Received 10.11.2015, received in revised form 15.01.2016, accepted 20.02.2016

*An algorithm is developed to determine coefficients of the stability polynomials such that the explicit Runge-Kutta methods have a predetermined shape and size of the stability region. Inequalities for accuracy and stability control are obtained. The impact of the stability control on efficiency of explicit methods to solving stiff problems is shown. Numerical calculations confirm that the three-step method of the first order with extended stability region is more efficient than the traditional three-stage method of the third order.*

*Keywords: stiff problem, explicit methods, stability region, accuracy and stability control.*

DOI: 10.17516/1997-1397-2016-9-2-209-219.

---

## Introduction

In some cases large-scale stiff problems need to be solved with the algorithms based on explicit methods as shown in [1, 2]. The  $L$ -stable methods suffer from the decomposition of the Jacobi matrix [3, 4] and in case of high dimension of a problem the calculation of the inverse to the Jacobi matrix defines overall computational efforts. At the same time, the algorithms based on explicit formulas are more efficient if stiffness of the problem allows to get an approximation to a solution in a reasonable time [5].

At the present time, algorithms of variable step and order are developed [1, 2, 6]. They use inequalities for stability control as a criterion for choosing between methods of high and low order of accuracy at the integration step. The numerical formulas of low-order methods are based on the same stages as high-order methods but its stability regions are much wider than stability regions of high-order ones. In a settling region there is no point in using high-order methods because the integration step is restricted by the condition of stability. The efficiency can be increased by applying some low-order methods with extended stability regions there. Further raising of efficiency may be achieved by developing not only variable order and step algorithms but algorithms with variable number of stages. We need some pool of low-order methods with extended stability regions to obtain them [1, 7].

The stability polynomials of degree up to  $m = 13$  are constructed in [1, 8]. Here an algorithm to determine the coefficients of the stability polynomial coefficients is developed such that the

---

\*novikov@icm.krasn.ru

†mixailrybkov@yandex.ru

© Siberian Federal University. All rights reserved

corresponding explicit Runge-Kutta methods have a predetermined shape and size of the stability region. Formulas for the coefficients of Runge-Kutta methods with extended stability regions are obtained. The stability and accuracy control inequalities for the first order methods are constructed. The three-stage explicit Runge-Kutta method is considered as an example. The results of comparison of the three-step method of the first order with extended stability region and the traditional three-stage method of the third order applied to solving stiff problems are given.

## 1. Explicit Runge-Kutta methods

We consider the Cauchy problem for the stiff system of ordinary differential equations

$$y' = f(t, y), \quad y(t_0) = y_0, \quad t_0 \leq t \leq t_k, \tag{1}$$

where  $y$  and  $f$  are real  $N$ -dimensional vector functions,  $t$  is an independent variable. In [9] the authors propose to solve (1) with explicit Runge-Kutta methods

$$y_{n+1} = y_n + \sum_{i=1}^m p_i k_i, \quad k_i = hf\left(t_n + \alpha_i h, y_n + \sum_{j=1}^{i-1} \beta_{ij} k_j\right), \tag{2}$$

where  $k_i$ ,  $1 \leq i \leq m$  are the stages of the method,  $h$  is an integration step,  $p_i$ ,  $\alpha_i$ ,  $\beta_{ij}$ ,  $1 \leq i \leq m$ ,  $1 \leq j \leq i - 1$  are numerical coefficients that define stability and accuracy characteristics of the scheme (2). For the sake of simplicity further we consider the Cauchy problem for the autonomous system of ordinary differential equations

$$y' = f(y), \quad y(t_0) = y_0, \quad t_0 \leq t \leq t_k. \tag{3}$$

For solving (3) we also may write formulas (2) in the following form:

$$y_{n+1} = y_n + \sum_{i=1}^m p_i k_i, \quad k_i = hf\left(y_n + \sum_{j=1}^{i-1} \beta_{ij} k_j\right). \tag{4}$$

The results given below can be used for non-autonomous systems if in (2) we assume

$$\alpha_1 = 0, \quad \alpha_i = \sum_{j=1}^{i-1} \beta_{ij}, \quad 2 \leq i \leq m.$$

## 2. Order conditions

Below we need matrix the  $B_m$  with elements  $b_{ij}$  of the form [1, 3]

$$\begin{aligned} b_{1i} &= 1, \quad 1 \leq i \leq m, \\ b_{ki} &= 0, \quad 2 \leq k \leq m, \quad 1 \leq i \leq k - 1, \\ b_{ki} &= \sum_{j=k-1}^{i-1} \beta_{ij} b_{k-1,j}, \quad 2 \leq k \leq m, \quad k \leq i \leq m, \end{aligned} \tag{5}$$

where  $\beta_{ij}$  are numerical coefficients of the scheme (2) or (4). The stability of one-step methods is usually investigated by applying a Runge-Kutta method to a linear scalar equation known as the Dahlquist equation [10, 11]

$$y' = \lambda y, \quad y(0) = y_0, \quad t \geq 0$$

with complex  $\lambda$ ,  $\Re(\lambda) < 0$ . The variable  $\lambda$  is considered as a certain eigenvalue of the Jacobi matrix of the problem (1) or (3). Applying numerical scheme (4) to solve the Dahlquist equation we get

$$y_{n+1} = Q_m(z)y_n, \quad Q_m(z) = 1 + \sum_{i=1}^m c_i z^i, \tag{6}$$

$$c_i = \sum_{j=1}^m b_{ij} p_j, \quad 1 \leq i \leq m,$$

where  $z = h\lambda$ . Denoting  $c = (c_1, \dots, c_m)^T$  and  $p = (p_1, \dots, p_m)^T$ , the latter equality (6) we can rewrite in the form

$$B_m p = c, \tag{7}$$

where the elements of the matrix  $B_m$  are defined in (5).

Expanding exact and approximate solutions in Taylor series in degrees of  $h$ , we get

$$y(t_{n+1}) = y(t_n) + hf + \frac{1}{2}h^2 f' f + \frac{1}{6}h^3 [f'' f^2 + f'^2 f] + O(h^4),$$

$$y_{n+1} = y_n + \left( \sum_{j=1}^m b_{1j} p_j \right) h f_n + \left( \sum_{j=2}^m b_{2j} p_j \right) h^2 f'_n f_n + \tag{8}$$

$$+ \left( \sum_{j=3}^m b_{3j} p_j \right) h^3 f''_n f_n + 0.5 \left( \sum_{j=2}^m b_{2j}^2 p_j \right) h^3 f''_n f_n^2 + O(h^4),$$

where elementary differentials are calculated on the exact  $y(t_n)$  and the approximate  $y_n$  solutions, respectively. If we compare equalities (8) assuming that  $y_n = y(t_n)$  we can see that the numerical scheme (4) has the first order of accuracy if

$$\sum_{j=1}^m b_{1j} p_j = 1$$

holds. Hence, we need to put  $c_1 = 1$  in the linear algebraic system (7) to construct  $m$ -stage methods of the first order. The remaining coefficients  $c_i$ ,  $2 \leq i \leq m$ , can be chosen according to the required stability characteristics. The stability function of an  $m$ -stage explicit Runge-Kutta method is known to be a polynomial of degree  $m$ . So we need the coefficients of this polynomial in order to use (7). Conversely, the coefficients of the polynomial can be calculated using (7) if the coefficients of the method (4) are given.

### 3. Stability polynomials

Let two integer number  $k$  and  $m$ ,  $k \leq m$  be given. Consider the polynomial

$$Q_{m,k}(x) = 1 + \sum_{i=1}^k c_i x^i + \sum_{i=k+1}^m c_i x^i, \tag{9}$$

where the coefficients  $c_i$ ,  $1 \leq i \leq k$ , are given and  $c_i$ ,  $k+1 \leq i \leq m$ , are free. The coefficients  $c_i$ ,  $1 \leq i \leq k$ , are usually defined from the approximation requirements. Therefore, to be precise, we assume  $c_i = 1/i!$ ,  $1 \leq i \leq k$  below.

Let  $|\gamma_m|$  denote the length of the stability interval of an  $m$ -stage explicit Runge-Kutta method, i.e. on the interval  $[\gamma_m, 0]$  the condition  $|Q_{m,k}(x)| \leq 1$  holds. It is easy to show that the more  $m$ , the faster coefficients of the stability polynomial considered on the interval

$[\gamma_m, 0]$  tend to zero [1]. Using the algorithm [1, 8] the coefficients  $c_i$ ,  $k+1 \leq i \leq m$ , are obtained up to the degree  $m = 13$ . Therefore, we consider an algorithm of obtaining the polynomials with prescribed properties on the interval  $[-1, 1]$ . In this case the coefficients  $c_i$  grow not so fast and it is possible to construct polynomials of degree  $m > 13$ . We transform the interval  $[\gamma_m, 0]$  to the interval  $[-1, 1]$  with the change of variables  $x = 1 - 2z/\gamma_m$  and get the polynomial

$$Q_m(z) = \sum_{i=0}^m d_i z^i. \quad (10)$$

Then, the coefficients  $d_i$ ,  $0 \leq i \leq m$ , of the polynomial (10) are related to the coefficients  $c_i$ ,  $0 \leq i \leq m$ , of the polynomial (9) by

$$c = UVd, \quad (11)$$

where  $d = (d_0, \dots, d_m)^T$ ,  $c = (c_0, \dots, c_m)^T$ ,  $U$  is a diagonal matrix with  $u^{ii} = (-2/\gamma_m)^{i-1}$ ,  $1 \leq i \leq m+1$ , on the main diagonal and the elements  $v^{ij}$  of the matrix  $V$  are given by the formulas

$$\begin{aligned} v^{1j} &= 1, \quad 1 \leq j \leq m+1, \\ v^{ij} &= v^{i,j-1} + v^{i-1,j-1}, \quad 2 \leq i \leq j \leq m+1, \\ v^{ij} &= 0, \quad i > j. \end{aligned}$$

It is easy to see that  $V$  is the Pascal triangle and its elements can be easily calculated by a recurrent formula. Having constructed the polynomial (10) on the interval  $[-1, 1]$ , we can calculate the coefficients of the polynomial (9) on the interval  $[\gamma_m, 0]$  using (11).

Now, turn to the polynomial (10). Let  $z_1, z_2, \dots, z_{m-1}$  denote the points of extremum of (10), assuming that  $z_1 > z_2 > \dots > z_{m-1}$ . We can find the coefficients  $d_i$ ,  $0 \leq i \leq m$ , such that the polynomial (10) assumes some prescribed values at the points of extremum  $z_i$ ,  $1 \leq i \leq m-1$ , i.e.

$$Q_m(z_i) = F_i, \quad 1 \leq i \leq m-1,$$

where  $F(z)$  is a prescribed function  $F_i = F(z_i)$ . For this purpose consider the algebraic system of equations in variables  $z_i$ ,  $1 \leq i \leq m-1$ , and  $d_i$ ,  $0 \leq i \leq m$ , of the form

$$\begin{aligned} Q_m(z_i) &= F_i, \quad Q'_m(z_i) = 0, \quad 1 \leq i \leq m-1; \\ Q'_m(z) &= \sum_{i=1}^m i d_i z^{i-1}, \end{aligned} \quad (12)$$

where the normalization requirements  $Q_m(-1) = (-1)^m$  and  $Q_m(1) = 1$  hold.

We rewrite (12) in the form that is convenient for computations. Let  $y$ ,  $w$  and  $g$  denote vectors with components

$$\begin{aligned} y_j &= z_j, \quad 1 \leq j \leq m-1; \quad w_i = d_{i-1}, \quad 1 \leq i \leq m+1; \\ g_i &= F_i, \quad 1 \leq i \leq m-1; \quad g_i = 1, \quad i = m; \quad g_i = (-1)^m, \quad i = m+1. \end{aligned}$$

Let  $E_1$  and  $E_2$  denote matrices of dimensions  $(m+1) \times (m+1)$  and  $(m-1) \times (m+1)$ , respectively, with the following elements on the main diagonal

$$e_1^{jj} = j-1, \quad 1 \leq j \leq m+1; \quad e_2^{ii} = 1/y_i, \quad 1 \leq i \leq m-1.$$

Let  $A$  denote an  $(m+1) \times (m+1)$  matrix with the following elements

$$a^{ij} = y_i^{j-1}, \quad 1 \leq i \leq m-1, \quad 1 \leq j \leq m+1;$$

$$a^{mj} = 1, \quad a^{m+1,j} = (-1)^{j+1}, \quad 1 \leq j \leq m+1.$$

Then, we can rewrite the problem (12) in the form

$$Aw - g = 0, \quad E_2 A E_1 w = 0. \quad (13)$$

We use the relaxation method for numerical solution of (13) [1, 12]. After obtaining the coefficients of the polynomial (10) we can calculate the coefficients of the polynomial (9) using (11). The value of  $\gamma_m$  is found from the condition that the desired polynomial corresponds to a first order method, i.e.  $c_1 = 1$ . If we take the second row from (13) and make necessary computations, we get

$$\gamma_m = -\frac{2}{c_1} \cdot \sum_{j=1}^{m+1} v_{2j} d_j.$$

## 4. Construction of stability regions

Numerical results [1, 8] show that the more  $m$ , the less the coefficient  $c_m$  of the polynomial (9) and, in particular, if  $m = 13$  and  $k = 1$  the value of  $c_m$  is about  $10^{-26}$ . Rounding errors make difficulties for computation of polynomial coefficients when  $m > 13$ . When solving the system (13), the coefficients  $d_i$ ,  $0 \leq i \leq m$ , of the polynomial (10) are established to grow in absolute value as  $m$  increases and if  $m = 13$  the maximum of  $|d_i|$  is about  $10^5$ . When  $m = 25$  we have the maximum of  $|d_i|$  to be about  $10^9$ , i.e. the coefficients  $d_i$  grow with lower speed. We can get the coefficients of the polynomial (9) from the coefficients of the polynomial (10) by formula (11) after solving the problem (13). It allows to calculate coefficients of stability polynomials up to  $m = 27$ .

The size and shape of stability region depend on the values of the function  $F$ . If we assume  $F_i = (-1)^i$ ,  $k \leq i \leq m+1$ , then the length of the stability interval is known to be  $|\gamma_m| = 2m^2$  [1]. In this case we have the maximal length of stability interval along the real axis for given  $m$ . Such stability regions are almost multiply connected and rounding errors can provoke appearing of small imaginary parts of eigenvalues of the Jacobi matrix so that stability region shrinks.

In order for rounding errors to not reduce the stability region, we need to "stretch" it along the imaginary axis in the points of extremum of the stability polynomial. For this purpose we assume  $F_i = (-1)^i \cdot \mu$ ,  $1 \leq i \leq m-1$ ,  $0 < \mu < 1$ . Calculations show that if we choose  $\mu = 0.9$ , then the length of the stability interval reduces by 5–8% in comparison with the maximal possible length that equals  $2m^2$ . In addition, such stability region stretches along the imaginary axis in the points of extremum. It provides better properties of stability of the method to rounding errors while stability region is reduced insignificantly. In case of  $\mu = 0.95$  the length of the stability interval reduces only by 3–4%. If  $\mu = 0.9$ , the stability region of the five-stage Runge-Kutta method has the shape shown in Fig. 1. The length of its stability interval is equal to  $|\gamma_m| = 46.79$ . For better visualization of the roots of polynomial (9) in complex plane all the figures include level lines  $|Q_{m,k}(x)| = q$ , if  $q$  is equal to 1, 0.8, 0.6, 0.4, and 0.2.

The less  $\mu$  from one to zero, the closer to each other the polynomial roots are situated on the real axis. So it is natural that the length of the stability interval of corresponding method decreases. Ellipses that are strongly pronounced at  $\mu = 1$  approach each other. Besides, it does not provide sufficiently large stretching along the imaginary axis. Therefore, depending on the characteristics of the problem to be solved, it is appropriate to use values of  $\mu$  in the interval between 0.8 and 0.95.

In case of the first-order methods, i.e. if  $k = 1$ , this condition can be met by choosing the values of the function  $F$ . For instance, if  $m = 4$ ,  $k = 1$ ,  $F = \{0.85, 0.95, 0.85\}$  we get the polynomial that meets this condition. Since  $m$  is even and all the values of  $F$  are positive, the graph of the polynomial does not cross the real axis, and the polynomial has two couples of

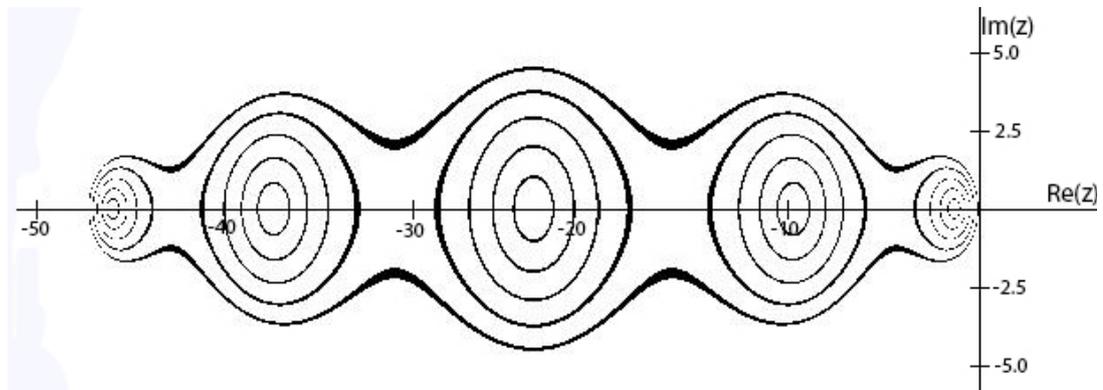


Fig. 1. The stability region for  $m = 5$ ,  $k = 1$ ,  $F = \{-0.9, 0.9, -0.9, 0.9\}$ ,  $|\gamma_m| = 46.79$

complex conjugate roots. That is why the stability region stretches along the imaginary axis in the complex plane and catches a part of it. At the same time the length of the stability interval along the real axis is not so big and equals  $|\gamma_m| = 2.18$ . The less the values of  $F$ , the more the length of the stability interval is, and, in particular, if  $F = \{0.55, 0.65, 0.55\}$  the stability region becomes practically rectangular with  $|\gamma_m| = 5.30$ . Further decreasing of the values of function  $F$  provides extension of the length of stability interval  $|\gamma_m|$ , but stability region catches a smaller part of the imaginary axis. Thus, when constructing the first-order methods for solving problems with oscillating solutions, it is reasonable to choose stability polynomials that have a couple of complex conjugate roots near the origin in the complex plane  $\{h\lambda\}$ . Besides, the values of function  $F$  that correspond to these roots should be chosen close to 1 in order that the stability region catches the maximal part of the imaginary axis.

If  $m$  is odd then at least one root of the polynomial (9) is real. We can use this in different ways. For example, if we place a real root of the polynomial of degree 5 between two couples of complex conjugate roots by choosing respective values of function  $F$ , then we can construct a stability region that is similar to rectangle. If the stability polynomial assumes the values  $F = \{0.2, 0.5, -0.5, -0.2\}$  at the points of extremum, then the stability region of the respective 5-stage method has the shape shown in Fig. 2. In the general case, we can construct the stability regions of different shape and size by choosing parameters  $m$ ,  $k$  and the values of function  $F$ .

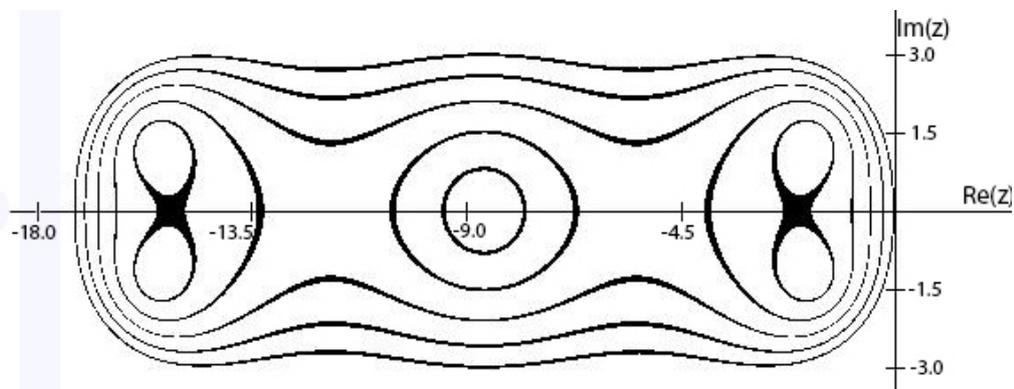


Fig. 2. The stability region for  $m = 5$ ,  $k = 1$ ,  $F = \{0.2, 0.5, -0.5, -0.2\}$ ,  $|\gamma_m| = 17.21$

## 5. Accuracy and stability control

Now we derive the inequality for accuracy and stability control of the first-order explicit Runge-Kutta methods. These schemes are supposed to be used in a settling region where an integration step is restricted by stability but not by accuracy of the method. Accuracy control is subsidiary there and to obtain the respective inequality we use the estimate of the local truncation error. Comparing Taylor expansions (8) of the exact solution  $y(t_{n+1})$  and the approximate solution  $y_{n+1}$  up to terms with  $h^2$ , provided that  $y_n = y(t_n)$ , we get that the local truncation error of the scheme (2) has the following form:

$$\delta_{n,1} = \left(0.5 - \sum_{i=2}^m b_{2i} p_i\right) h^2 f' f + O(h^3).$$

Using (5), it can be rewritten in the form

$$\delta_{n,1} = 0.5(1 - 2c_2)h^2 f' f + O(h^3),$$

where  $c_2$  is the coefficient at  $x^2$  of the stability polynomial (9).

The value of  $\delta_{n,1}$  can be estimated using stages  $k_i$ ,  $1 \leq i \leq m$ , that have already been calculated in many ways. We use the following notation:

$$A'_n = g'_{m1} \|k_2 - k_1\|, \quad A''_n = g''_{m1} \|hf(y_{n+1}) - k_1\|,$$

where

$$g''_{m1} = |1 - 2c_{m2}|/2, \quad g'_{m1} = g''_{m1}/|\beta_{21}|.$$

Then, taking into account that

$$k_2 - k_1 = \beta_{21} h^2 f'_n f_n + O(h^3),$$

$$hf(y_{n+1}) - k_1 = h^2 f'_n f_n + O(h^3),$$

we can use the following inequalities to control accuracy and choose the integration step, respectively

$$A'_n \leq \varepsilon, \quad A''_n \leq \varepsilon. \quad (14)$$

The vector  $k_1$  depends linearly on the step size. So using the first inequality of (14) the repeated computation of the solution is accompanied with just one extra calculation of the right hand side of the system of differential equations (1). If the integration step is successful, then the second inequality does not provide the growth of computational efforts because the vector  $f(y_{n+1})$  is used only at the next integration step. If the second inequality of (14) does not hold, then the return will be expensive in terms of computational efforts. Besides, the more  $m$ , the higher computational efforts. Nevertheless, preliminary control of  $A'_n$  usually allows to avoid extra computations.

Now we derive the inequality for stability control by the method described in [6]. We write the stages  $k_1$ ,  $k_2$  and  $k_3$  for the problem  $y' = Ay$ , where  $A$  is a matrix with constant coefficients. As a result, we get

$$k_1 = X \cdot y_n, \quad k_2 = (X + \beta_{21} X^2) \cdot y_n,$$

$$k_3 = [X + (\beta_{31} + \beta_{32})X^2 + \beta_{21}\beta_{32}X^3] \cdot y_n,$$

where  $X = hA$ . We find the coefficients  $s_1$ ,  $s_2$ , and  $s_3$  assuming that the following equality

$$s_1 k_1 + s_2 k_2 + s_3 k_3 = X^3 \cdot y_n$$

holds. It is easy to see that this requirement holds on the conditions that

$$s_1 = \frac{\beta_{31} + \beta_{32} - \beta_{21}}{\beta_{21}^2 \beta_{32}}, \quad s_2 = -\frac{\beta_{31} + \beta_{32}}{\beta_{21}^2 \beta_{32}}, \quad s_3 = \frac{1}{\beta_{21} \beta_{32}}.$$

It is also easy to see that the equality

$$\frac{1}{\beta_{21}}(k_2 - k_1) = X^2 \cdot y_n.$$

holds. Then, according to [6] the maximal eigenvalue  $v_n = h\lambda_{n \max}$  of the Jacobi matrix of the system (1) can be estimated by the power iterations. I.e. we use the following inequality for stability control of the first-order methods (2):

$$v_n \leq |\gamma_{m,1}|, \quad v_n = \beta_{21} \cdot \max_{1 \leq i \leq N} \left| \frac{s_1 k_1^i + s_2 k_2^i + s_3 k_3^i}{k_2^i - k_1^i} \right|, \quad (15)$$

where the positive constants  $\gamma_{m,1}$  depend on the size of the stability regions of numerical schemes. Inequality (15) can be used on each step for choosing efficient numerical scheme.

## 6. First-order method

The problem of construction of first-order explicit Runge-Kutta methods with given stability regions comes to solving the system of linear algebraic equations (7) with nondegenerate matrix  $B_m$ . In this case the values of the vector  $c = (c_1, c_2, \dots, c_m)^T$  define the size and the shape of the stability region. As an example, we consider 3-stage explicit Runge-Kutta method in the form

$$\begin{aligned} y_{n+1} &= y_n + p_1 k_1 + p_2 k_2 + p_3 k_3, \\ k_1 &= hf(y_n), \quad k_2 = hf(y_n + \beta_{21} k_1), \quad k_3 = hf(y_n + \beta_{31} k_1 + \beta_{32} k_2). \end{aligned} \quad (16)$$

Assume that  $\beta_{21} = 0.5$ ,  $\beta_{31} = -1$  and  $\beta_{32} = 2$ . Then, on each step the increments  $k_1$ ,  $k_2$  and  $k_3$  are calculated at the points  $t_n$ ,  $t_n + 0.5h$  and  $t_n + h$ , respectively. The numerical results show that calculation of stages at these points increase the reliability of calculations when integrating stiff problems.

The coefficients  $p_1$ ,  $p_2$  and  $p_3$  are obtained by means of equality (7), where the vector of the right part of the system includes coefficients of the stability polynomial of the method. These are calculated by solving the problems (11) and (13). If we assume  $F_i = 0.95 \cdot (-1)^i$ ,  $1 \leq i \leq 2$ , then the length of the stability interval of the method decreases insignificantly in comparison with the maximal possible length  $2m^2$  that equals 17.46. The values of the coefficients of the stability polynomial are  $c_1 = 1$ ,  $c_2 = 0.15209292726978$  and  $c_3 = 0.00580524400854$  for the given  $F_i$ . And the stability region is simply connected in this case. The requirement of the first order leads to the equalities

$$p_1 + p_2 + p_3 = 1, \quad \beta_{21} p_2 + (\beta_{31} + \beta_{32}) p_3 = c_2, \quad \beta_{21} \beta_{32} p_3 = c_3.$$

As a result, we have coefficients

$$p_3 = \frac{c_3}{\beta_{21} \beta_{32}}, \quad p_2 = \frac{c_2 - (\beta_{31} + \beta_{32}) p_3}{\beta_{21}}, \quad p_1 = 1 - p_2 - p_3.$$

We use inequalities (14) for the accuracy control of the numerical schemes of the first order. The length of the stability interval of the first-order method equals 17.46. So we can use inequality  $v_3 \leq 17.46$  where  $v_3$  is defined in formula (15) for the stability control of this scheme.

## 7. Third-order method

For numerical solution of (3) we consider an explicit three-stage Runge-Kutta method in the form

$$\begin{aligned} y_{n+1} &= y_n + r_1 k_1 + r_2 k_2 + r_3 k_3, \\ k_1 &= hf(y_n), \quad k_2 = hf(y_n + \beta_{21} k_1), \\ k_3 &= hf(y_n + \beta_{31} k_1 + \beta_{32} k_2), \end{aligned} \quad (17)$$

where the coefficients  $\beta_{21}$ ,  $\beta_{31}$  and  $\beta_{32}$  are defined in the description of the first-order method and the coefficients  $r_1$ ,  $r_2$  and  $r_3$  need to be defined. The conditions of the third order of accuracy are the following

$$r_1 + r_2 + r_3 = 1, \quad \frac{1}{2}r_2 + r_3 = \frac{1}{2}, \quad \frac{1}{4}r_2 + r_3 = \frac{1}{3}, \quad 2r_3 = \frac{1}{3}.$$

As a result, we have the coefficients of the scheme (17) of the third order

$$\beta_{21} = \frac{1}{2}, \quad \beta_{31} = -1, \quad \beta_{32} = 2, \quad r_1 = \frac{1}{6}, \quad r_2 = \frac{2}{3}, \quad r_3 = \frac{1}{6}.$$

We derive the accuracy control inequality for a third-order method using the idea of embedded methods. We consider an auxiliary scheme  $y_{n+1,1} = y_n + k_2$  that has the second order of accuracy. Then, the accuracy control inequality is the following [6]

$$\|y_{n+1} - y_{n+1,1}\| = \|k_1 - 2k_2 + k_3\|/6 \leq \varepsilon,$$

where  $\|\cdot\|$  is some norm in  $R^N$ ,  $\varepsilon$  is the required accuracy of calculations.

## 8. Numerical results

The calculations were performed on AMD A6-3420M APU with double precision. The required accuracy was  $\varepsilon = 10^{-2}$ . In concrete computations the norm  $\|\xi_n\|$  from the inequalities for the accuracy control was calculated by the formula

$$\|\xi_n\| = \max_{1 \leq i \leq N} |\xi_n^i| / (|y_n^i| + r),$$

where  $i$  is the number of a vector component,  $r$  is a positive parameter. If the inequality  $|y_n^i| < r$  holds on the component with the number  $i$ , then the absolute error  $r \cdot \varepsilon$  is controlled, otherwise we control the relative error  $\varepsilon$ . During calculations the parameter  $p$  was chosen in a way that the practical accuracy was not worse than the required one.

We compared the efficiency of two methods. The first one is the Runge-Kutta method of the first order with accuracy and stability control described in section 6. And the second one is the traditional three-stage Runge-Kutta method of the third order with accuracy control. Below numbers  $is$ ,  $iw$  and  $if$  denote the total number of integration steps, the total number of repeated solution calculations (returns) because of required accuracy violation and the total number of calculations of the right hand side of the problem (1), respectively.

As a test example, we consider simplified model with periodic solution that corresponds to a Belousov-Zhabotinsky reaction (oregonator) in the form [14]

$$y_1' = 77.27(y_2 - y_1 y_2 + y_1 - 8.375 \cdot 10^{-6} y_1^2), \quad y_2' = (-y_2 - y_1 y_2 + y_3)/77.27, \quad y_3' = 0.161(y_1 - y_3),$$

$$t \in [0, 300], \quad y_1(0) = y_3(0) = 4, \quad y_2(0) = 1.1, \quad h_0 = 10^{-3}.$$

The given problem is "too" stiff for explicit methods. Nevertheless, we present exactly this example in order to emphasize advantages of explicit methods with stability control.

Using the first-order method the computational efforts are  $is = 427\,018$ ,  $iw = 16\,149$  and  $if = 1\,725\,219$ . For the third-order method respective computational efforts are  $is = 2\,903\,722$ ,  $iw = 769\,200$  are  $if = 10\,249\,566$ . The results show fivefold benefit of the first-order method with extended stability region in comparison with the traditional method of the third order. Both decreasing of number of returns because of stability control and extending the stability region lead to the growth of efficiency. At the end of the integration interval the practical accuracy is better than the required one for all the algorithms. This tendency holds when solving other examples [14].

## Conclusion

The coefficients of stability polynomials of degree up to  $m = 27$  and the respective methods of the first order are obtained using the algorithm described above. The maximal length of the stability interval of  $m$ -stage first-order Runge-Kutta method is known to be  $2m^2$ . So each computation of the right part of the system corresponds to  $2m$  units of the stability interval. This means that if the integration step is restricted by stability of the scheme then the more  $m$ , the more efficient method becomes. Using the inequality for stability control does not lead to increase of computational efforts because the estimation of the maximal eigenvalue of the Jacobi matrix of the system (1) is calculated via stages computed in advance and it does not lead to the growth of number of calculations of function  $f$ . Thus, the first-order methods with extended stability regions allow to significantly increase the efficiency in a settling region where the step is restricted by stability.

The developed set of the first-order methods with stability and accuracy control can be applied in the algorithms with variable step, variable order and variable number of stages. In this case the efficiency increases not only because of switching from the method of high order into the method of low order in a settling region, but by means of "boosting" an integration step in this region using the first-order methods with extended stability regions.

*This work is partially supported by Russian Foundation of Basic Research (grant 14-01-00047).*

## References

- [1] E.A Novikov, Explicit methods for stiff systems, Novosibirsk, Nauka, 1997 (in Russian).
- [2] E.A.Novikov, Yu.V.Shornikov, Computer simulation of stiff hybrid systems, Novosibirsk, Izdat. NSTU, 2012 (in Russian).
- [3] E.Hairer, S.P.Norsett, G.Wanner, Solving ordinary differential equations. I. Nonstiff problems, Berlin, Springer, 1987.
- [4] E.Hairer, G.Wanner, Solving ordinary differential equations. II. Stiff and differential-algebraic problems, Berlin, Springer, 1996.
- [5] V.A.Novikov, E.A.Novikov, Raising the efficiency of algorithms for the integration of ordinary differential equations at the expense of loss of stability, *USSR Computational Mathematics and Mathematical Physics*, **25**(1985), no. 4, 39–43.
- [6] E.A.Novikov, Variable order and step algorithm based on a stages of Runge–Kutta method of third order of accuracy, *Izv. Saratov Univ. (N.S.), Ser. Mat. Meh. Inform.*, **11**(2011), no. 3(1), 46–53 (in Russian).

- [7] E.A.Novikov, M.V Rybkov, The numerical algorithm of constructing of stability regions for explicit methods, *Sistemy upravleniya i informatsionye tehnologii*, **55**(2014), no. 1.1, 173–177 (in Russian).
- [8] E.A.Novikov, The construction of the stability regions for the explicit Runge-Kutta methods, *Vychislitelnye metody i programirovanie*, **10**(2009), 248–257 (in Russian).
- [9] V.A.Novikov, E.A.Novikov, Control of the stability of explicit one-step methods of integrating ordinary differential equations, *Soviet Math. Dokl.*, **30**(1984), №1, 211–215.
- [10] G.Dahlquist, A special stability problem for linear multistep methods *BIT*, **3**(1963), 23–43.
- [11] K.Dekker, J.G.Verwer, *Stability of Runge-Kutta methods for stiff nonlinear differential equations*, Amsterdam, North Holland, 1984.
- [12] G.Korn, T.Korn, *Mathematical handbook*, Moscow, Nauka, 1970 (in Russian).
- [13] A.E.Novikov, E.A.Novikov, L-stable (2,1)-method for stiff nonautonomius problem solving, *Vychislitelnye tehnologii, Tom. 13, Vestnik KazNU*, **58**(2008), no. 3, 477–482 (in Russian).
- [14] W.H.Enright, T.E.Hull, Comparing numerical methods for the solutions of stiff systems of ODE's, *BIT*, **15**(1975), 10–48.

## Применение явных методов с расширенными областями устойчивости для решения жестких задач

**Евгений А. Новиков**

Институт вычислительного моделирования СО РАН  
Академгородок, 50/44, Красноярск, 660036

Россия

**Михаил В. Рыбков**

Институт математики и фундаментальной информатики  
Сибирский федеральный университет  
Свободный, 79, Красноярск, 660041

Россия

---

*Разработан алгоритм определения коэффициентов полиномов устойчивости, при которых явные методы типа Рунге-Кутты имеют заданные форму и размер области устойчивости. Получены неравенства для контроля точности и устойчивости численных формул первого порядка. Исследовано влияние контроля устойчивости на эффективность явных методов применительно к решению жестких задач. Приведены результаты расчетов, показывающие повышение эффективности трехстадийного метода первого порядка с расширенной областью устойчивости в сравнении с традиционным трехстадийным методом третьего порядка.*

*Ключевые слова: жесткая задача, явные методы, область устойчивости, контроль точности и устойчивости.*